CLAIMS

What is claimed is:

1.   1.   A method of automatically generating a keystream segment of an arbitrary location of
2.   a complete keystream of an additive stream cipher, the method comprising the
3.   computer-implemented steps of:
4.   receiving a location value that identifies a location of the keystream segment within
5.   the complete keystream;
6.   creating and storing a state value for a leaf node of a balanced binary tree, wherein the
7.   tree represents the complete keystream and the leaf node represents the
8.   keystream segment at the location, by a preorder traversal of the tree from root
9.   node to the leaf node wherein a leftward tree branch transition comprises
10.   computing a first non-linear function and a rightward tree branch transition
11.   comprises computing a second non-linear function;
12.   creating and storing the keystream segment by applying a third function to the state
13.   value of the leaf node.

1.   2.   A method as recited in Claim 1, further comprising the steps of creating and storing
2.   the balanced binary tree by creating and storing a stack of h elements wherein the ith
3.   element of said stack stores a state datum for the ith node on a path from a root node of
4.   the tree to the leaf node.

1.   3.   A method as recited in Claim 2, wherein the step of creating and storing a state value
2.   for a leaf node comprises the steps of computing and storing a state value for the leaf
3.   node that is unique with respect to any other state value that is computed at any other
4.   time for any other leaf node of the tree.

50325-0512 (Seq. No. 3245)

1   4.   A method as recited in Claim 2, wherein the steps of computing a first non-linear

2   function and computing a second non-linear function comprises the steps of

3   computing first and second non-linear functions that are selected such that a set of all

4   state values of all leaf nodes is indistinguishable from a random value.

1   5.   The method as recited in Claim 1, wherein each leaf node stores n bits of state

2   information, wherein n is a multiple of four.

1   6.   The method as recited in Claim 1, further comprising the steps of:

2   creating and storing $3n$ bits of state information in each leaf node comprising a

3   concatenation of three $n/2$ bit quantities $z\,|\,y\,|\,x$, wherein n is a multiple of

4   four;

5   computing the first non-linear function a and the second non-linear function b as the

6   composition of a diffusion function d with the nonlinear "confusion" functions

7   f and g, wherein $a = f \circ d$ and $b = g \circ d$ and wherein

8   $f(\,z\,|\,y\,|\,x) = 2z\,|\,S\,(R\,(S\,(R\,(y)\,)\,)\,)\,|\,L\,(S\,(L\,(S\,(x)\,)\,)\,)$

9   $g(\,z\,|\,y\,|\,x) = 2z+1\,|\,L\,(S\,(L\,(S\,(y)\,)\,)\,)\,)\,|\,S\,(R\,(S\,(R\,(\Box x)\,)\,)\,)$

10   $d(\,z\,|\,y\,|\,x) = z\;|\;x + y + z\;|\;2x + y + z$

11   $c(z\,|\,y\,|\,x) = x \oplus y$

12   wherein integer addition modulo two is denoted as +, bitwise exclusive-or is denoted

13   as $\oplus$, and bitwise complementation is denoted as $\Box$;

14   wherein the R denotes rotation by $n/4$ bits to in a direction of a least significant bit and

15   L denotes rotation by $n/4$ bits in a direction of a most significant bit; and

16   wherein a nonlinear function S comprises a lookup in a key-dependent substitution

17   table.

1   7.   The method as recited in Claim 1, wherein the third function comprises computing a

2   linear reduction of n bits of the state value to $n/2$ bits thereof.

50325-0512 (Seq. No. 3245)

1   8.   A method as recited in Claim 6, wherein the third function comprises computing a

2   bitwise Boolean exclusive OR of x and y.

1   9.   A method as recited in Claim 6, further comprising the steps of creating and storing

2   the substitution table S by selecting four invertible functions and storing the four

3   invertible functions in a concatenated form.

1   10.   A method as recited in Claim 6, further comprising the steps of computing functions f

2   and g in seven instructions of a central processing unit that can issue two instructions

3   simultaneously, by using five registers to store values of x, y, z, a temporary variable,

4   and a pointer to the substitution table S.

1   11.   A method as recited in Claim 6, wherein the substitution table S comprises an array of

2   randomly selected integer values.

1   12.   A method as recited in Claim 6, wherein the substitution table S comprises an array of

2   256 randomly selected 32-bit unsigned integer values.

1   13.   The method as recited in Claim 1, further comprising the steps of creating and storing

2   a key for use by the first non-linear function and the second non-linear function,

3   wherein the key comprises a table of randomly selected values.

1   14.   The method as recited in Claim 1, further comprising the steps of creating and storing,

2   once and at a time prior to receiving the location value, a key for use by the first non-

3   linear function and the second non-linear function, wherein the key comprises a table

4   of randomly selected values.

-27-

1   15.   The method as recited in Claim 1, further comprising the steps of creating and storing

2   a key in the form of a plurality of pseudo-randomly selected invertible functions,

3   wherein each of the invertible functions maps an 8-bit portion of the state value to an

4   8-bit quantity for use as a substitute portion of the state value.

1   16.   A method as recited in Claim 1, wherein the substitution table S comprises a plurality

2   of sub-tables, and wherein generating the substitution table comprises (a) setting

3   values of the sub-tables to key-dependent permutations and (b) setting values of one

4   of the sub-tables to an exclusive OR of itself to the identity permutation.

5   17.   A method of enciphering a plaintext using at least one keystream segment at an

6   arbitrary location of a complete keystream, the method comprising the computer-

7   implemented steps of:

8   receiving a segment of a plaintext;

9   receiving a location value that identifies a location of the keystream segment within

10   the complete keystream;

11   creating and storing a state value for a leaf node of a balanced binary tree, wherein the

12   tree represents the complete keystream and the leaf node represents the

13   keystream segment at the location, by a preorder traversal of the tree from root

14   node to the leaf node wherein a leftward tree branch transition comprises

15   computing a first non-linear function and a rightward tree branch transition

16   comprises computing a second non-linear function;

17   creating and storing the keystream segment by applying a third function to the state

18   value of the leaf node;

19   enciphering the segment of the plaintext by combining the keystream segment with

20   the segment of the plaintext using a Boolean exclusive OR operation to result

21   in creating and storing a segment of ciphertext.

-28-

18. A method of encrypting an ordered plurality of packets of a network communication link using at least one keystream segment at an arbitrary location of a complete keystream, the method comprising the computer-implemented steps of:

receiving a packet from among the plurality of packets;

determining a location value that represents a relative location of the packet among the plurality of packets;

creating and storing a state value for a leaf node of a balanced binary tree, wherein the tree represents the complete keystream and the leaf node represents a keystream segment at the relative location, by a preorder traversal of the tree from root node to the leaf node wherein a leftward tree branch transition comprises computing a first non-linear function and a rightward tree branch transition comprises computing a second non-linear function;

creating and storing the keystream segment by applying a third function to the state value of the leaf node;

enciphering the packet by combining the keystream segment with data of the packet using a Boolean exclusive OR operation to result in creating and storing enciphered packet data.

19. A computer-readable medium carrying one or more sequences of instructions for automatically generating a keystream segment of an arbitrary location of a complete keystream of an additive stream cipher, which instructions, when executed by one or more processors, cause the one or more processors to carry out the steps of:

receiving a location value that identifies a location of the keystream segment within the complete keystream;

creating and storing a state value for a leaf node of a balanced binary tree, wherein the tree represents the complete keystream and the leaf node represents the keystream segment at the location, by a preorder traversal of the tree from root node to the leaf node wherein a leftward tree branch transition comprises computing a first non-linear function and a rightward tree branch transition comprises computing a second non-linear function;

-29-

13 creating and storing the keystream segment by applying a third function to the state
14   value of the leaf node.

1

1 20. An apparatus for automatically generating a keystream segment of an arbitrary
2   location of a complete keystream of an additive stream cipher, comprising:
3   means for receiving a location value that identifies a location of the keystream
4   segment within the complete keystream;
5   means for creating and storing a state value for a leaf node of a balanced binary tree,
6   wherein the tree represents the complete keystream and the leaf node
7   represents the keystream segment at the location, by a preorder traversal of the
8   tree from root node to the leaf node wherein a leftward tree branch transition
9   comprises computing a first non-linear function and a rightward tree branch
10   transition comprises computing a second non-linear function;
11   means for creating and storing the keystream segment by applying a third function to
12   the state value of the leaf node.

1 21. An apparatus for automatically generating a keystream segment of an arbitrary
2   location of a complete keystream of an additive stream cipher, comprising:
3   a network interface that is coupled to the data network for receiving one or more
4   packet flows therefrom;
5   a processor;
6   one or more stored sequences of instructions which, when executed by the processor,
7   cause the processor to carry out the steps of:
8   receiving a location value that identifies a location of the keystream segment
9   within the complete keystream;

50325-0512 (Seq. No. 3245)

10   creating and storing a state value for a leaf node of a balanced binary tree,

11   wherein the tree represents the complete keystream and the leaf node

12   represents the keystream segment at the location, by a preorder

13   traversal of the tree from root node to the leaf node wherein a leftward

14   tree branch transition comprises computing a first non-linear function

15   and a rightward tree branch transition comprises computing a second

16   non-linear function;

17   creating and storing the keystream segment by applying a third function to the

18   state value of the leaf node.

50325-0512 (Seq. No. 3245)